Guía Examen.

**WPF:**

CLASE CONEXIÓN:

class Conexion

{

SqlCommand cmd;

SqlDataReader rd;

//MÉTODO PARA CONECTAR LA BASE DE DATOS

public static SqlConnection conectarBd()

{

SqlConnection cnn;

try

{

cnn = new SqlConnection("Data Source = LUCALIZARDIAD3F; Initial Catalog = datosClientes; User ID = sa; Password = sqladmin");

cnn.Open();

}

catch (Exception e)

{

cnn = null;

MessageBox.Show("ERROR AL CONECTAR" + e);

}

return cnn;

}

//Método para llenar el comboBox cuando dependa de algo

public void llenarComboNombres(ComboBox cb,String pais)

{

cb.Items.Clear();

try

{

SqlConnection con;

con = Conexion.conectarBd();

cmd = new SqlCommand(String.Format("select nombre from cliente where pais like'{0}'", pais), con);

rd = cmd.ExecuteReader();

while (rd.Read())

{

cb.Items.Add(rd["nombre"].ToString());

}

cb.SelectedIndex = 0;

rd.Close();

con.Close();

}

catch (Exception ex)

{

MessageBox.Show("no se pudo llenar el combo" + ex);

}

}

//Método para llenar el comboBox sin que dependa de algo

public void llenarComboPais(ComboBox cb)

{

try

{

SqlConnection con;

con = Conexion.conectarBd();

cmd = new SqlCommand("select distinct pais from cliente", con);

rd = cmd.ExecuteReader();

while (rd.Read())

{

cb.Items.Add(rd["pais"].ToString());

}

cb.SelectedIndex = 0;

rd.Close();

con.Close();

}

catch (Exception ex)

{

MessageBox.Show("no se pudo llenar el combo" + ex);

}

}

}

//Método que busca una clave

public int buscaClave(String nombre)

{

int res = 0;

SqlConnection con = Conexion.conectarBd();

SqlCommand cmd = new SqlCommand(String.Format("select idCliente from cliente where nombre='{0}'", nombre), con);

SqlDataReader rd = cmd.ExecuteReader();

if (rd.Read())

{

res = rd.GetInt32(0);

rd.Close();

}

return res;

}

//modificar

public int modificar(Int32 idCliente, double saldoNuevo)

{

int res = 0;

//abrir la conexion

SqlConnection con;

con = Conexion.conectarBd();

//command para ejecutar el query (insert)

SqlCommand cmd2 = new SqlCommand(String.Format("update transaccion set saldo ={0} where idCliente ={1}", saldoNuevo, idCliente), con);

res = cmd2.ExecuteNonQuery(); //num de colunas(registros) afectados (-1 indica no modificaciones)

//cerrar la conexion

con.Close();

return res;

}

//Método para agregar una actividad.sin ID

public String Alta(Curso c)

{

String res = "";

int i = 0;

try

{

SqlConnection miConexion = Conexion.conectarBd();

SqlCommand cmd = new SqlCommand("select top(1) idCurso from curso order by idCurso desc", miConexion);

SqlDataReader rd = cmd.ExecuteReader();

if (rd.Read())

{

i = rd.GetInt32(0) + 1;

rd.Close();

SqlCommand cmd2 = new SqlCommand(String.Format("insert into curso values ({0},'{1}',{2},{3})", i,c.nombre,c.horas,c.idEstado), miConexion);

cmd2.ExecuteNonQuery();

}

else

{

i = rd.GetInt32(0) + 1;

rd.Close();

SqlCommand cmd2 = new SqlCommand(String.Format("insert into curso values ({0},'{1}',{2},{3})", i, c.nombre, c.horas, c.idEstado), miConexion);

}

}

catch (Exception ex)

{

res = "Alta no exitosa: " + ex.Message;

}

return res;

}

web:

\*NO ES NECESARIA LA CLASE CONEXIÓN

conectar la bd

protected OdbcConnection conectarBD()

{

String stringConexion = "Driver={SQL Server Native Client 11.0};Server=CC102-05\\sa;Uid=sa;Pwd=adminadmin;Database=gameSpot";

try

{

OdbcConnection conexion = new OdbcConnection(stringConexion);

conexion.Open();

lbContador.Text = "conexion exitosa";

return conexion;

}

catch (Exception ex)

{

lbContador.Text = ex.StackTrace.ToString();

return null;

}

}

LLENAR UN DDLIST

public void llenarDd(DropDownList dd)

{

try

{

OdbcConnection con;

con = conectarBD();

OdbcCommand cmd = new OdbcCommand("select nombre from trabajador", con);

OdbcDataReader rd = cmd.ExecuteReader();

while (rd.Read())

{

dd.Items.Add(rd["nombre"].ToString());

}

//cb.SelectedIndex = 0;

rd.Close();

}

catch (Exception ex)

{

}

}